AMENDMENTS TO THE CLAIMS

Please cancel claims 37 and 50-65 without prejudice. Kindly amend claims 31, 34, and 38 as shown in the following listing of claims. The listing of claims will replace all prior versions, and listings, of claims in the application:

Listing of Claims

1-30. (canceled)

31. (currently amended) A pipelined microprocessor comprising:

an instruction cache that is indexed by a fetch address, said instruction cache for caching instructions, and for providing said instructions to an instruction buffer for storage therein, wherein said instructions comprise[[s]] variable byte-length instructions;

a branch target address cache, coupled to said instruction buffer and indexed by said fetch address, for caching branch target addresses of previously executed branch instructions;

said instruction buffer comprising an indicator associated with each byte of each of said instructions stored in said instruction buffer, wherein said indicator has a true value if said branch target address cache predicts that said byte is an opcode byte of one of said instructions and that said one of said instructions is one of said previously executed branch instructions and the microprocessor has speculatively branched to one of said branch target addresses cached for said one of said previously executed branch instructions.

32-33. (canceled)

34. (currently amended) A method of speculatively branching in a pipelined microprocessor, comprising:

caching, in a branch target address cache (BTAC), a plurality of branch target addresses of previously executed branch instructions and a bit associated with each of said branch instructions, wherein said bit is true only if the associated branch instruction spans more than one instruction cache line, wherein the microprocessor processes variable byte-length instructions;

accessing said BTAC with a fetch address of an instruction cache after said caching;

determining whether said fetch address hits in said BTAC in response to said accessing; ~~and~~

branching the microprocessor to one of said plurality of branch target addresses selected by said fetch address if said fetch address hits in said BTAC whether or not a branch instruction is cached in a line of said instruction cache indexed by said fetch address;

storing in an instruction buffer instructions provided by said instruction cache selected by said fetch address, and storing a discrete indication for each byte of each said instruction, wherein said discrete indication is true if said BTAC predicts said byte is an opcode byte of said instruction and said branching the microprocessor to one of said plurality of branch target addresses was performed for said instruction.

35. (original) The method of claim 34, further comprising:

storing a branch direction prediction associated with each of said plurality of branch target addresses prior to said accessing said BTAC.

36. (original) The method of claim 35, wherein said branching the microprocessor to said one of said plurality of branch target addresses selected by said fetch address is performed only if said associated branch direction prediction indicates said branch instruction will be taken.

37. (canceled)

38. (currently amended) The method of claim ~~37~~34, further comprising:

determining from said discrete indication, subsequent to said storing, that said branching was performed.

39-42. (canceled)

43. (previously presented) The microprocessor of claim 31, further comprising:

instruction decode logic, coupled to said instruction buffer, for decoding said instructions to indicate which byte of each of said instructions is an opcode byte; and

prediction check logic, coupled to receive said indicator associated with each byte of said instructions from said instruction buffer, wherein if one of said indicators associated with one of said instructions specifies the microprocessor has speculatively branched to said one of said branch target addresses of said one of said instructions, said prediction check logic indicates that the microprocessor erroneously branched to said one of said branch target addresses if said byte of said one of said instructions associated with said one of said indicators is not indicated by said instruction decode logic to be said opcode byte.

44. (previously presented) The microprocessor of claim 31, further comprising:

instruction decode logic, coupled to said instruction buffer, for decoding said instructions to indicate whether each of said instructions is a non-branch instruction; and

prediction check logic, coupled to receive said indicator associated with each byte of said instructions from said instruction buffer, wherein if one of said indicators associated with one of said instructions specifies the microprocessor has speculatively branched to said one of said branch target addresses of said one of said instructions, said prediction check logic indicates that the microprocessor erroneously branched to said one of said branch target addresses if said instruction decode logic indicates said one of said instructions is a non-branch instruction.

45. (previously presented) The microprocessor of claim 31, wherein said branch target address cache is further configured to cache a length of each of said previously executed branch instructions, the microprocessor further comprising:

instruction decode logic, coupled to said instruction buffer, for determining a length of each of said instructions; and

prediction check logic, coupled to receive said indicator associated with each byte of said instructions from said instruction buffer, wherein if one of said indicators associated with one of said instructions specifies the microprocessor has speculatively branched to said one of said branch target addresses of said one of said previously executed branch instructions, said prediction check logic indicates that the microprocessor erroneously branched to said one of said branch target addresses if said length received from said instruction decode logic does not match said length of said one of said previously executed branch instructions provided by said branch target address cache.

46. (previously presented) The microprocessor of claim 31, further comprising:

prediction check logic, coupled to receive said indicator associated with each byte of said instructions from said instruction buffer, wherein said one of said instructions is a branch instruction, wherein if one of said indicators associated with said branch instruction specifies the microprocessor has speculatively branched to said one of said branch target addresses of said branch instruction, said prediction check logic indicates that the microprocessor erroneously branched to said one of said branch target addresses if a resolved direction of said branch instruction does not match a direction of said branch instruction predicted by said branch target address cache.

47. (previously presented) The microprocessor of claim 31, further comprising:

prediction check logic, coupled to receive said indicator associated with each byte of said instructions from said instruction buffer, wherein said one of said instructions is a branch instruction, wherein if one of said indicators associated with said branch instruction specifies the microprocessor has speculatively branched to said one of said branch target addresses of said branch instruction, said prediction check logic indicates that the

microprocessor erroneously branched to said one of said branch target addresses if a resolved target address of said branch instruction does not match said one of said branch target addresses to which the microprocessor speculatively branched.

48. (previously presented) The microprocessor of claim 31, further comprising:

a non-speculative branch predictor, coupled to said instruction buffer, for generating a non-speculative predicted target address of a branch instruction for which said branch target address cache provided said one of said branch target addresses to which the microprocessor speculatively branched; and

branch control logic, coupled to receive said indicator associated with each byte of said instructions from said instruction buffer, wherein if one of said indicators associated with said branch instruction specifies the microprocessor has speculatively branched to said one of said branch target addresses of said branch instruction, said branch control logic causes the microprocessor to branch to said non-speculative predicted target address if said non-speculative predicted target address generated by said non-speculative branch predictor does not match said one of said branch target addresses of said branch instruction provided by said branch target address cache.

49. (previously presented) The microprocessor of claim 31, further comprising:

a non-speculative branch predictor, coupled to said instruction buffer, for generating a non-speculative predicted direction of a branch instruction for which said branch target address cache provided said one of said branch target addresses to which the microprocessor speculatively branched; and

branch control logic, coupled to receive said indicator associated with each byte of said instructions from said instruction buffer, wherein if one of said indicators associated with said branch instruction specifies the microprocessor has speculatively branched to said one of said branch target addresses of said branch instruction, said branch control logic

causes the microprocessor to branch to a next instruction sequential to said branch instruction if said non-speculative predicted direction generated by said non-speculative branch predictor is a not taken prediction.

50-70. (canceled)